Finite Horizon Trading Strategy with Transaction Costs and Exponential Utility in a Regime Switching Market

XU Shanghua

Supervisor : Prof. DAI Min

An academic exercise presented in partial fulfilment for the degree of Master of Science in Mathematics

Department of Mathematics

National University of Singapore

August 2010

Abstract

This thesis studies the finite horizon optimal trading strategy with proportional transaction costs in a regime switching stock market. This problem is an extension of the classic investment strategy in a static economic condition. The exponential utility function is considered here. The study of this problem is mainly motivated by Dai et. al. (2010), in which the finite horizon optimal investment problem with proportional transaction costs under logarithm utility function in a regime switching market is studied. In this thesis, we use dynamic programming approach to derive the Hamilton-Jacobi-Bellman (HJB) equations satisfied by the value functions. For our exponential utility case, the transformation is different from the one in the logarithm utility case, and we will get a system of variational inequalities with gradient constraints. For the power utility case, there is also a similar system with gradient constraints. The difference lies in that the case with exponential utility cannot lead to a self-contained system of double obstacle problems. Due to the fact that no closed-form solution exists, we employ two numerical methods, namely

the penalty method and the projected SOR method to solve the system of variational inequalities based on certain assumptions. Finally we show the optimal trading strategies.

List of Author's Contributions

The author has proposed two numerical algorithms to solve the finite horizon optimal investment problem with proportional transaction costs in a regime switching market (under exponential utility), for which no analytical solution exists yet. The transformation from the original 3-dimension problem to the 2-dimension problem is presented. Although the problem with gradient constraints is not easy to solve, we show that the system of variational inequalities cannot be transformed into a self-contained system of double obstacle problems. So we need to be faced with the gradient constraints. The results of two numerical algorithms are equivalent. The numerical results can also explain some phenomena in economics. For example, we will see that younger investors are more sensitive to changes in the rate of return of risky asset than elder ones.

Acknowledgement

I would like to thank my supervisor, Prof. Dai Min, for his suggestions and guidance over the past two years. With his help and the teaching in the modules of financial modeling & modeling and numerical simulations, I have learnt many numerical algorithms through some projects which make me possible to do research in financial mathematics and to finish my thesis. What I have learnt from him will benefit me in the future.

I would also like to express my thanks to my family, my lecturers and my seniors Wang Shengyuan, Li Peifan and Zhao Kun for their guidance throughout my life. I sincerely appreciate all your help along the way.

Contents

Chapter 1

Introduction

1.1 Historical Work

In this paper, the optimal trading strategies for an exponential utility investor who faces proportional transaction costs are studied. This is an extension of the classic investment strategy in a static economic condition.

The study of portfolio optimization problems via stochastic processes in continuous time was initiated by Merton (1969). He formulated the investment problem in infinite time horizon, and extended the model to finite time horizon. The investor chooses how to allocate his funds between investment in a risk-free asset (' bank account ') and a risky asset (' stock ') in order to maximize the expected utility of terminal wealth over a finite horizon. In the absence of transaction costs, the optimal strategy would be time-independent under certain assumptions, and it is to keep a constant fraction (' Merton proportion ') of total wealth in the risky asset. However, such a strategy will lead to incessant trading, which is impracticable in the real world.

The proportional transaction costs model was first introduced by Magill and Constantinides (1976), and it leads to a stochastic singular control problem. They provided a heuristic argument that the optimal strategy is described by a no-transaction region, which means the investor does not buy or sell stocks unless his portion of wealth in stock moves out of this region. Since then, there have been a lot of papers studying the optimal trading strategies for an investor facing proportional transaction costs. When the investor's horizon is infinite, the strategy is simplified since it is time-independent.

However, the finite horizon portfolio selection problem with proportional transaction costs has remained unsolved until recently. Liu and Loewenstein (2002) approximated the strategy by a sequence of analytical solutions that converge to the real solution. Dai and Yi (2009) characterized the strategy by PDE approach. They proved that the original HJB equation is equivalent to a double-obstacle problem. Uichanco (2006) used the penalty method to solve the obstacle problem and found it to be more efficient. At the same time, researchers have started to consider the portfolio selection problem with regime switching feature, which means that the economic condition switches stochastically between two market conditions. Jang et. al. (2007) considered an infinite horizon problem in a bull-bear switching market and explained the puzzle of liquidity premium. Dai et. al. (2010) considered a finite horizon portfolio selection problem with transaction costs in a regime switching market in order to study the issue of leverage management. This paper is largely motivated by the success of the approaches applied to the optimal investment problem in the regime switching market in the above two papers.

1.2 Scope of this paper

In this paper, we propose numerical solutions to solve the finite horizon optimal investment problem with proportional transaction costs in a regime switching market. There is only one risky asset, the price of which follows the geometric Brownian motion. Similar arguments as in Dai et. al. (2010) will be used to derive the HJB equations satisfied by the investor's value function in each regime, and exponential utility function will be studied. The HJB equation leads to a system of variational inequalities with gradient constraints which correspond to the optimal buying and selling boundaries. The system of variational inequalities cannot be transformed into a double-obstacle problem as in Dai et. al. (2010). The penalty method and the projected SOR method will be employed to numerically solve the variational inequalities. To compare the results, we plot the optimal buying and selling boundaries obtained from both approaches. We will also examine the effects of varying parameters such as the transaction costs proportion.

The rest of the paper is organized as follows. In Chapter 2, we present the formulation of the model. Then in Chapter 3, we discuss the transformation differences between exponential utility function and logarithm utility function. In Chapter 4, we propose numerical algorithms to solve the problems raised in Chapter 2 and 3. We show the numerical results and analyses in Chapter 5. The paper ends with a conclusion in Chapter 6.

Chapter 2

Model Formulation

In this chapter, we consider the finite horizon portfolio selection problem with proportional transaction costs in a regime switching market. Our model formulation follows that of Dai et. al. (2010).

2.1 The asset market

The financial market under consideration consists of two assets: a riskless asset, referred to as the bank account, and a risky asset, referred to as a stock. Their price processes, denoted by P_t and Q_t respectively, are assumed to satisfy:

$$dP_t = r(\varepsilon_t)P_t dt,$$

$$dQ_t = Q_t[\alpha(\varepsilon_t)dt + \sigma(\varepsilon_t)dB_t],$$

where $\varepsilon_t \in \{1,2\}$ denotes the changing market condition that switches between two regimes, "bull market" (regime 1) and "bear market" (regime 2), which is governed by a two-state Markov chain with generators

$$\begin{pmatrix} k_1 & -k_1 \\ -k_2 & k_2 \end{pmatrix},$$

where $k_1, k_2 > 0$. In other word, regime i switches into regime j at the first jump time of an independent Poisson process with intensity k_i, for $i \neq j \in \{1,2\}$.

For $i = 1,2$, we assume that $r(i) > 0, \alpha(i) > r(i)$, and $\sigma(i) > 0$ are constants representing the risk free interest rate, the expected rate of return and the volatility of the stock respectively in regime i. The process $\{B(t): t \geq 0\}$ is a standard Brownian motion, independent of ε_t, on a filtered probability space $(\Omega, F, \{F_t\}_{t\geq 0}, P)$ with $B_0 = 0$ almost surely. We denote $r_i = r(i)$, $\alpha_i = \alpha(i)$, and $\sigma_i = \sigma(i)$ later on.

Let X_t and Y_t denote the monetary value of the investor's holdings at time t, in the bank account and stock respectively. With the assumption of proportional transaction costs, X_t and Y_t evolve according to the following equations in regime i:

$$dX_t = r_i X_{t-} dt - (1+\lambda) dL_t + (1-\mu) dM_t, \tag{2.1}$$

$$dY_t = \alpha_i Y_{t-} dt + \sigma_i Y_{t-} dB_t + dL_t - dM_t. \tag{2.2}$$

where L_t and M_t are right-continuous (with left hand limits), nonnegative, and nondecreasing $\{F_t\}_{t\geq 0}$ adapted processes with $L_0 = M_0 = 0$, representing cumulative dollar values up to time t for the purpose of buying and selling stock respectively. The constants $\lambda \in [0,+\infty)$ and $\mu \in [0,1)$ represent the proportional transaction costs incurred on buying and selling of stock respectively. We further assume $\lambda + \mu > 0$ to ensure the presence of transaction costs. From (2.1) and (2.2), it can be noted that the purchase of dL_t worth of stock involves a payment of $(1+\lambda)dL_t$ from the bank account while the sale of dM_t worth of stock realizes only $(1-\mu)dM_t$ in cash.

2.2 The investor's problem

The investor's net wealth at time t, denoted by W_t, is defined as the monetary value of the holdings in the bank account after selling off all shares of the stock. Notice the assumption $\alpha_i > r_i$ implies that it is never optimal for the investor to short sale the stock and as a result we always have $Y_t \geq 0$. Due to transaction costs, we have:

$$W_t = X_t + (1-\mu)Y_t .$$

The solvency region S is defined to be

$$S = \{(x,y) \in R^2 : y > 0\} .$$

Assume that the investor is given an initial position (x,y) in S. His problem is to choose an investment strategy (L,M) in the admissible investment strategies $A_s(x,y)$, which ensures that (X_t, Y_t) given by (2.1) and (2.2) is in the solvency region S for all $t \in [s,T)$.

Given an initial position of $(x_{0,}y_0) \in S$, the investor's problem is to choose an admissible strategy so as to maximize the expected utility of terminal wealth, that is, to maximize $E_0^{x_0,y_0}[U(W_T)]$. Here $E_t^{x,y}$ denotes the conditional expectation at time t given the initial endowment $X_t = x$, $Y_t = y$. Moreover, we assume that the investor has a utility function given by:

$$U(W) = -e^{-\beta W}, \beta > 0$$

The value function in regime $i \in \{1,2\}$ is defined to be

$$\varphi_i(x,y,t) = \sup_{(L,M)\in A_t(x,y)} E_t^{x,y}[U(W_T)], (x,y) \in S, t \in [0,T]. \qquad (2.3)$$

2.3 HJB equation

The main point of this paper is not the rigorous mathematical derivation, but the numerical algorithms to solve the problem, so we only present a heuristic derivation of the optimality equation satisfied by the value function as given in (2.3). We first consider a restricted class of policies in which L_t and M_t are constrained to be absolutely continuous with bounded derivatives, i.e.

$$L_t = \int_0^t l_s ds, \ M_t = \int_0^t m_s ds, \ 0 < l_s, m_s \le \kappa$$

The Bellman equation governing the value function φ_i is

$$\max_{l,m}\{\partial_t\varphi_i + r_i x\partial_x\varphi_i + \alpha_i y\partial_y\varphi_i + \frac{1}{2}\sigma_i^2 y^2\partial_{yy}\varphi_i + l[\partial_y\varphi_i - (1+\lambda)\partial_x\varphi_i]$$

$$+ m[(1-\mu)\partial_x\varphi_i - \partial_y\varphi_i] - k_i(\varphi_i - \varphi_j)\} = 0, \text{ for } i \ne j \in \{1,2\}. \qquad (2.4)$$

Maximization with respect to l, m will produce a solution given by

$$l^* = \begin{cases} \kappa & \text{if} \quad \partial_y \varphi_i - (1+\lambda)\partial_x \varphi_i \geq 0 \\ 0 & \text{if} \quad \partial_y \varphi_i - (1+\lambda)\partial_x \varphi_i < 0 \end{cases}$$

$$m^* = \begin{cases} \kappa & \text{if} \quad (1-\mu)\partial_x \varphi_i - \partial_y \varphi_i \geq 0 \\ 0 & \text{if} \quad (1-\mu)\partial_x \varphi_i - \partial_y \varphi_i < 0 \end{cases}$$

The above solution is similar to the infinite horizon optimal portfolio selection problem studied by Davis and Norman (1990). This indicates that in each regime i, the optimal trading strategies are to buy or sell at the maximum rate or not at all. The solvency region S is divided into three regions, "Buying" (BR_i), "Selling" (SR_i) and "No Transaction" (NT_i). At the boundary between the BR_i and NT_i regions, $\partial_y \varphi_i = (1+\lambda)\partial_x \varphi_i$, while at the boundary between the SR_i and NT_i regions, $(1-\mu)\partial_x \varphi_i = \partial_y \varphi_i$.

Thus, (2.4) can be rewritten as

$$\begin{cases} \partial_t \varphi_i + L_i \varphi_i + \kappa[\partial_y \varphi_i - (1+\lambda)\partial_x \varphi_i]^+ + \kappa[(1-\mu)\partial_x \varphi_i - \partial_y \varphi_i]^+ = 0 \\ \varphi_i(x,y,T) = -e^{-\beta(x+(1-\mu)y)}, (x,y) \in S, t \in [0,T) \end{cases}$$

where

$$L_i \varphi_i = \frac{1}{2}\sigma_i^2 y^2 \partial_{yy} \varphi_i + \alpha_i y \partial_y \varphi_i + r_i x \partial_x \varphi_i - k_i(\varphi_i - \varphi_j),$$

for $i \neq j \in \{1,2\}$.

Letting $\kappa \to \infty$, we obtain the equation satisfied by the original value function:

$$\begin{cases} \min\{-\partial_t \varphi_i - L_i \varphi_i, -(1-\mu)\partial_x \varphi_i + \partial_y \varphi_i, (1+\lambda)\partial_x \varphi_i - \partial_y \varphi_i\} = 0 \\ \varphi_i(x,y,T) = -e^{-\beta(x+(1-\mu)y)}, (x,y) \in S, t \in [0,T) \end{cases}, \qquad (2.5)$$

for $i \neq j \in \{1,2\}$.

Chapter 3

Differences between Exponential Utility and Logarithm Utility: The Transformation Problem

3.1 Dimension reduction: 3 to 2

Equation (2.5) is a 3-dimension problem, which is dimension-reducible. For the logarithm utility function $U(W) = \log W$, we can use the homogeneity to deduce that for any positive constant ρ,

$$\varphi_i(\rho x, \rho y, t) = \varphi_i(x, y, t) + \log \rho$$

Therefore, by using the following transformation:

$$z = \frac{x}{y}$$

$$V_i(z, t) = \varphi_i(\frac{x}{y}, 1, t) = \varphi_i(x, y, t) - \log y$$

$$\partial_t \varphi_i = \partial_t V_i$$

$$\partial_x \varphi_i = \frac{\partial_z V_i}{y}$$

$$\partial_y \varphi_i = \frac{1 - z \partial_z V_i}{y}$$

$$\partial_{yy}\varphi_i = \frac{z^2\partial_{zz}V_i + 2z\partial_z V_i - 1}{y^2}$$

(2.5) can be reduced to:

$$\begin{cases} \min\{-\partial_t V_i - L_i^* V_i, -(z+1-\mu)\partial_z V_i + 1, (z+1+\lambda)\partial_z V_i - 1\} = 0 \\ V_i(z,T) = \log(z+1-\mu) \end{cases} \text{in } \Omega_T \quad (3.1)$$

where $\Omega_T = (\mu - 1, +\infty) \times [0,T)$, and

$$L_i^* V_i = \frac{1}{2}\sigma_i^2 z^2 \partial_{zz} V_i - (\alpha_i - r_i - \sigma_i^2) z \partial_z V_i + (\alpha_i - \frac{1}{2}\sigma_i^2) - k_i (V_i - V_j),$$

for $i \neq j \in \{1,2\}$.

However, for the exponential utility function, we have

$$\varphi_i(\rho x, \rho y, t) = -(-\varphi_i(x,y,t))^\rho$$

We can see that the dimension cannot be reduced in the above way.

In fact, for exponential utility function, we need to use a different transformation. For simplicity, we assume $r_1 = r_2 = r$. The essence of the correct transformation is to rule out the dependence of x, and relies on the fact that x becomes $e^{r(T-t)}x$ at maturity. The transformation is as following:

$$z = \beta y e^{r(T-t)}$$

$$\varphi_i = -e^{-\beta x e^{r(T-t)} - V_i(z,t)}$$

$$\partial_t \varphi_i = \varphi_i (r\beta x e^{r(T-t)} - \partial_t V_i + rz\partial_z V_i)$$

$$\partial_x \varphi_i = -\beta \varphi_i e^{r(T-t)}$$

$$\partial_y \varphi_i = -\beta \varphi_i e^{r(T-t)} \partial_z V_i$$

$$\partial_{yy} \varphi_i = \varphi_i (\beta e^{r(T-t)} \partial_z V_i)^2 - \varphi_i \beta^2 e^{2r(T-t)} \partial_{zz} V_i$$

Finally, (2.5) can be reduced to:

$$\begin{cases} \min\{-\partial_t V_i - L_i' V_i, \partial_z V_i - (1-\mu), (1+\lambda) - \partial_z V_i\} = 0 \\ V_i(z,T) = (1-\mu)z \end{cases} \quad \text{in } \Omega_T \tag{3.2}$$

where $\Omega_T = (0,+\infty)\times[0,T)$, and

$$L_i' V_i = (\alpha_i - r)z\partial_z V_i + \frac{1}{2}\sigma_i^2 z^2 \partial_{zz} V_i - \frac{1}{2}\sigma_i^2 z^2 (\partial_z V_i)^2 + k_i(1 - e^{V_i - V_j})$$

for $i \neq j \in \{1,2\}$. This is a system of variational inequalities with gradient constraints.

3.2 Trading regions

In each regime i, the "Buying" (BR_i), "Selling" (SR_i) and "No Transaction" (NT_i) regions are defined as following:

$$BR_i = \{(z,t) \in \Omega_T : \partial_z V_i(z,t) = 1+\lambda\}$$

$$SR_i = \{(z,t) \in \Omega_T : \partial_z V_i(z,t) = 1-\mu\}$$

$$NT_i = \{(z,t) \in \Omega_T : 1-\mu < \partial_z V_i(z,t) < 1+\lambda\}$$

Comparing to the solutions of (3.2), we are more interested in the buying and selling boundaries, which tell us how to trade at every time step in practice. But the above definition of the three regions does not show obviously the properties of the buying and selling boundaries. This is due to the difficulty in dealing with the variational inequalities with gradient constraints.

Dai et. al. (2010) has established an equivalence between (3.1) and a double-obstacle problem in logarithm utility case. Notice that (3.1) could be written in the following form:

$$\begin{cases} -\partial_t V_i - L_i^* V_i = 0, \dfrac{1}{z+1+\lambda} < \partial_z V_i < \dfrac{1}{z+1-\mu} \\ -\partial_t V_i - L_i^* V_i \geq 0, \partial_z V_i = \dfrac{1}{z+1+\lambda} \text{or} \dfrac{1}{z+1-\mu} \\ V_i(z,T) = \log(z+1-\mu) \end{cases} \tag{3.3}$$

in $\Omega_T = (\mu-1,+\infty)\times[0,T)$. Denote $v_i(z,t) = \partial_z V_i(z,t)$, we have

$$\partial_z(L_i^* V_i) = \frac{1}{2}\sigma_i^2 z^2 \partial_{zz} v_i - (\alpha_i - r_i - 2\sigma_i^2) z \partial_z v_i - (\alpha_i - r_i - \sigma_i^2) v_i - k_i(v_i - v_j)$$

$$=: L_i'' v_i .$$

It has been shown in Dai et. al. (2010) that (3.3) is equivalent to the following double-obstacle problem:

$$\begin{cases} -\partial_t v_i - L_i'' v_i = 0, \dfrac{1}{z+1+\lambda} < v_i < \dfrac{1}{z+1-\mu} \\ -\partial_t v_i - L_i'' v_i \geq 0, v_i = \dfrac{1}{z+1+\lambda} \\ -\partial_t v_i - L_i'' v_i \leq 0, v_i = \dfrac{1}{z+1-\mu} \\ v_i(z,T) = \dfrac{1}{z+1-\mu}, z \in (\mu-1,+\infty), t \in [0,T) \end{cases} \tag{3.4}$$

for $i \neq j \in \{1,2\}$.

However, for (3.2), a technical difficulty prevents us from transforming the problem into a double-obstacle problem. Actually, if we want to transform the gradient constraints, we need to use the transformation $v_i(z,t) = \partial_z V_i(z,t)$. Then we have

$$\partial_z(L_i' V_i) = (\alpha_i - r)(v_i + z\partial_z v_i) + \frac{1}{2}\sigma_i^2(2z\partial_z v_i + z^2 \partial_{zz} v_i) - \frac{1}{2}\sigma_i^2(2z v_i^2 + 2z^2 v_i \partial_z v_i)$$
$$- k_i e^{V_i - V_j}(v_i - v_j)$$

And we still cannot eliminate V_i and V_j. So we need to solve (3.2) directly. We will propose numerical algorithms to solve it in the next chapter.

Chapter 4

Numerical Schemes

In this chapter, we shall propose the numerical algorithms to solve (3.2). Due to the difficulty in dealing with these original variational inequalities with gradient constraints, we adopt both the penalty method and the projected SOR method.

4.1 The penalty method

Inspired by Uichanco (2006) and Dai and Zhong (2009), we use the penalty method to deal with the system (3.2). Then we have the following form:

$$\begin{cases} -\partial_t V_i - L_i^{'} V_i = l(1-\mu-\partial_z V_i)^+ + m(\partial_z V_i - 1 - \lambda)^+ \\ V_i(z,T) = (1-\mu)z \end{cases} \tag{4.1}$$

where $(z,t) \in (0,+\infty) \times [0,T)$, and $L_i^{'} V_i$ is given in (3.2), for $i \neq j \in \{1,2\}$. l, m are penalty parameters that can be chosen to be sufficiently large to ensure that $1-\mu-\varepsilon \le \partial_z V_i \le 1+\lambda+\varepsilon$, for any given $\varepsilon > 0$, $\varepsilon << 1$.

Boundary Conditions

In order to apply any implicit finite difference scheme, it is necessary to prescribe boundary conditions on the parabolic boundary. For this transformed problem, when z is large enough, the investor is in the selling region and hence should sell the stock. So we need to impose an upper bound M for z. In the following, we solve the problem in $(z,t) \in (0, M] \times [0,T)$. When $z = M$, we have $\partial_z V_i = 1 - \mu$. And when $z \to 0$, $\partial_z V_i = 1 + \lambda$.

Discretization

Here, we illustrate the use of the fully implicit scheme to solve the nonlinear partial differential equation (4.1) using the penalty method. Since i can take value 1 or 2, we need to solve a system of PDE. In Chong (2006), a similar discretization scheme was used to solve the finite horizon optimal investment problem in a static economic condition. We will now discretize equation (4.1) using the fully implicit scheme with upwind treatment to ensure diagonal dominance. We will use n and j to denote the indexes of the grid points in spatial direction and time direction respectively.

Denote the step size in space variable z and time variable t by dz and dt respectively. Then $z_n = ndz$ and $t_j = jdt$. Let $V_{1_{n,j}}^k$ and $V_{2_{n,j}}^k$ be the k-th step discrete solutions to (4.1) in Newton iteration at the point

(z_n, t_j) of regime 1 and 2 respectively. Thus we have

$$\begin{cases}
-\dfrac{V_{1_{n,j+1}} - V_{1_{n,j}}^{k+1}}{dt} - L_{z1}^{'} V_{1_{n,j}}^{k+1} - k_1 (1 - e^{V_{1_{n,j}}^{k} - V_{2_{n,j}}^{k}}) \\
= l(1 - \mu - \dfrac{V_{1_{n,j}}^{k+1} - V_{1_{n-1,j}}^{k+1}}{dz})^{+} + m(\dfrac{V_{1_{n+1,j}}^{k+1} - V_{1_{n,j}}^{k+1}}{dz} - 1 - \lambda)^{+} \\
-\dfrac{V_{2_{n,j+1}} - V_{2_{n,j}}^{k+1}}{dt} - L_{z2}^{'} V_{2_{n,j}}^{k+1} - k_2 (1 - e^{V_{2_{n,j}}^{k} - V_{1_{n,j}}^{k}}) \\
= l(1 - \mu - \dfrac{V_{2_{n,j}}^{k+1} - V_{2_{n-1,j}}^{k+1}}{dz})^{+} + m(\dfrac{V_{2_{n+1,j}}^{k+1} - V_{2_{n,j}}^{k+1}}{dz} - 1 - \lambda)^{+}
\end{cases}$$

(4.2)

with terminal condition

$$\begin{cases}
V_{1_{n,\frac{T}{dt}}} = (1-\mu) z_n \\
V_{2_{n,\frac{T}{dt}}} = (1-\mu) z_n
\end{cases}$$

where

$$L_{z1}^{'} V_{1_{n,j}}^{k+1} = (\beta_1 + \beta_2) V_{1_{n-1,j}}^{k+1} + (\beta_3 - 2\beta_1 - \beta_2) V_{1_{n,j}}^{k+1} + (\beta_1 - \beta_3) V_{1_{n,j}}^{k+1},$$

$$\beta_1 = -\frac{\sigma_1^2 n^2}{2},$$

$$\beta_2 = -\frac{\sigma_1^2 n^2}{2} (V_{1_{n,j}}^{k} - V_{1_{n-1,j}}^{k}),$$

$$\beta_3 = (\alpha_1 - r) n,$$

and

$$L_{z2}^{'} V_{2_{n,j}}^{k+1} = (\beta_1^{'} + \beta_2^{'}) V_{2_{n-1,j}}^{k+1} + (\beta_3^{'} - 2\beta_1^{'} - \beta_2^{'}) V_{2_{n,j}}^{k+1} + (\beta_1^{'} - \beta_3^{'}) V_{2_{n,j}}^{k+1},$$

$$\beta_1^{'} = -\frac{\sigma_2^2 n^2}{2},$$

$$\beta_2^{'} = -\frac{\sigma_2^2 n^2}{2} (V_{2_{n,j}}^{k} - V_{2_{n-1,j}}^{k}),$$

$$\beta_3' = (\alpha_2 - r)n.$$

We need to linearize the non-linear terms on the right hand side of (4.2).

For the terms $(1-\mu-\frac{V_{1_{n,j}}^{k+1}-V_{1_{n-1,j}}^{k+1}}{dz})^+$, $(\frac{V_{1_{n+1,j}}^{k+1}-V_{1_{n,j}}^{k+1}}{dz}-1-\lambda)^+$ and $(1-\mu-\frac{V_{2_{n,j}}^{k+1}-V_{2_{n-1,j}}^{k+1}}{dz})^+$, $(\frac{V_{2_{n+1,j}}^{k+1}-V_{2_{n,j}}^{k+1}}{dz}-1-\lambda)^+$, we use the generalized Newton iteration which has been used by Forsyth and Vetzal (2002) to solve the American option pricing problem. We get the following linearizations:

$$(1-\mu-\frac{V_{1_{n,j}}^{k+1}-V_{1_{n-1,j}}^{k+1}}{dz})^+ = (1-\mu-\frac{V_{1_{n,j}}^{k+1}-V_{1_{n-1,j}}^{k+1}}{dz})\times I_{\{1-\mu>\frac{V_{1_{n,j}}^{k}-V_{1_{n-1,j}}^{k}}{dz}\}}$$

$$(\frac{V_{1_{n+1,j}}^{k+1}-V_{1_{n,j}}^{k+1}}{dz}-1-\lambda)^+ = (\frac{V_{1_{n+1,j}}^{k+1}-V_{1_{n,j}}^{k+1}}{dz}-1-\lambda)\times I_{\{\frac{V_{1_{n+1,j}}^{k}-V_{1_{n,j}}^{k}}{dz}>1+\lambda\}}$$

$$(1-\mu-\frac{V_{2_{n,j}}^{k+1}-V_{2_{n-1,j}}^{k+1}}{dz})^+ = (1-\mu-\frac{V_{2_{n,j}}^{k+1}-V_{2_{n-1,j}}^{k+1}}{dz})\times I_{\{1-\mu>\frac{V_{2_{n,j}}^{k}-V_{2_{n-1,j}}^{k}}{dz}\}}$$

$$(\frac{V_{2_{n+1,j}}^{k+1}-V_{2_{n,j}}^{k+1}}{dz}-1-\lambda)^+ = (\frac{V_{2_{n+1,j}}^{k+1}-V_{2_{n,j}}^{k+1}}{dz}-1-\lambda)\times I_{\{\frac{V_{2_{n+1,j}}^{k}-V_{2_{n,j}}^{k}}{dz}>1+\lambda\}}$$

Substituting into (4.2), we get the systems of linear equations. We can use Gaussian elimination to solve the systems backward in time.

4.2 The projected SOR method

To deal with variational inequalities, we can also use the projected SOR method. (3.2) can be written as the following form:

$$[(-\partial_t - L_i^{'})V_i](\partial_z V_i - 1 + \mu)(\partial_z V_i - 1 - \lambda) = 0,$$

$$(-\partial_t - L_i^{'})V_i \ge 0,\ \partial_z V_i - 1 + \mu \ge 0,\ \partial_z V_i - 1 - \lambda \le 0, \tag{4.3}$$

$$V_i(z,T) = (1-\mu)z.$$

Following the discretization in the penalty method above, we have

$$(1 + (\beta_3 - 2\beta_1 - \beta_2)dt)V_{1_{n,j}}^{k+1} + (\beta_1 + \beta_2)dtV_{1_{n-1,j}}^{k+1} + (\beta_1 - \beta_3)dtV_{1_{n+1,j}}^{k+1}$$
$$\ge V_{1_{n,j+1}} + k_1(1 - e^{V_{1_{n,j}}^k - V_{2_{n,j}}^k})dt$$
$$(1 + (\beta_3^{'} - 2\beta_1^{'} - \beta_2^{'})dt)V_{2_{n,j}}^{k+1} + (\beta_1^{'} + \beta_2^{'})dtV_{2_{n-1,j}}^{k+1} + (\beta_1^{'} - \beta_3^{'})dtV_{2_{n+1,j}}^{k+1}$$
$$\ge V_{2_{n,j+1}} + k_2(1 - e^{V_{2_{n,j}}^k - V_{1_{n,j}}^k})dt$$

It can be written in matrix form:

$$(D_j - L_j - U_j)V_{1_j}^{k+1} \ge V_{1_{j+1}} + k_1(1 - e^{V_{1_j}^k - V_{2_j}^k})dt$$

$$(D_j^{'} - L_j^{'} - U_j^{'})V_{2_j}^{k+1} \ge V_{2_{j+1}} + k_2(1 - e^{V_{2_j}^k - V_{1_j}^k})dt$$

where D, L, U stand for the diagonal part, the negative lower triangular part and the negative upper triangular part of the matrix. Then we can use the following iterations:

$$u_{1_j}^{k+1} = (D_j - L_j)^{-1}[U_j V_{1_j}^k + V_{1_{j+1}} + k_1(1 - e^{V_{1_j}^k - V_{2_j}^k})dt]$$

$$V_{1_{n,j}}^{k+1} = \min\{\max\{V_{1_{n,j}}^k + \omega(u_{1_{n,j}}^{k+1} - V_{1_{n,j}}^k), V_{1_{n-1,j}}^{k+1} + (1-\mu)dz\}, V_{1_{n-1,j}}^{k+1} + (1+\lambda)dz\}$$

$$u_{2_j}^{k+1} = (D_j^{'} - L_j^{'})^{-1}[U_j^{'} V_{2_j}^k + V_{2_{j+1}} + k_2(1 - e^{V_{2_j}^k - V_{1_j}^k})dt]$$

$$V_{2_{n,j}}^{k+1} = \min\{\max\{V_{2_{n,j}}^k + \omega(u_{2_{n,j}}^{k+1} - V_{2_{n,j}}^k), V_{2_{n-1,j}}^{k+1} + (1-\mu)dz\}, V_{2_{n-1,j}}^{k+1} + (1+\lambda)dz\}$$

where $\omega \in (1,2)$ is a constant.

Using the projected SOR method, we only need to do iterations at each time step, and the gradient constraints are put into the comparison conditions. So we do not have to consider the penalty terms, and we can solve the linear equation systems by SOR approach. The general discretization scheme, boundary conditions and terminal conditions are the same as those in the penalty method.

Eventually, we will need to plot the buying and selling boundaries for problem (3.2). The numerical results will be shown in the next chapter.

Chapter 5

Numerical Results

5.1 The results of penalty method

We plot the optimal buying and selling boundaries in both bull market and bear market as functions of t using the above discretization in penalty method. Note that z here refers to the product of the parameter in the exponential utility function and what the current monetary value in the risky asset will be at maturity under risk free interest rate. This is different from the logarithm utility case, where z refers to the ratio of bank account holdings to holdings in the risky asset. In our case, since holdings in the risky asset are always assumed to be positive, the buying and selling boundaries are also positive, implying that the investor should never short sell stocks. But we do not know whether the bank account holdings are positive. If the investor follows exponential utility, then he or she does not need to consider whether to leverage, but just to follow the strategies which indicate the risky asset holdings. Once z falls below the buying boundary, the investor should buy stocks to bring the position back into the no transaction region. On the other hand, if z goes above the selling boundary, the investor should sell some of the stocks.

However, in both bull market and bear market, there exist a threshold value of t beyond which no buying boundary exists. This is consistent with the observation in Liu and Loewenstein (2002) for the finite horizon optimal investment problem. They found that the optimal fraction of wealth invested in stock decreases as time goes toward maturity because of the finite time horizon and proportional transaction costs.

We fix the following set of parameters: $r = 0.06$, $\alpha_1 = 0.2$, $\sigma_1 = 0.2$, $\alpha_2 = 0.1$, $\sigma_2 = 0.4$, $k_1 = 0.5$, $k_2 = 2.5$, $\lambda = 0.005$, $\mu = 0.01$, $T = 5$, $M = 4$, $l = m = 10^5$, and change the step size in time and spatial directions.

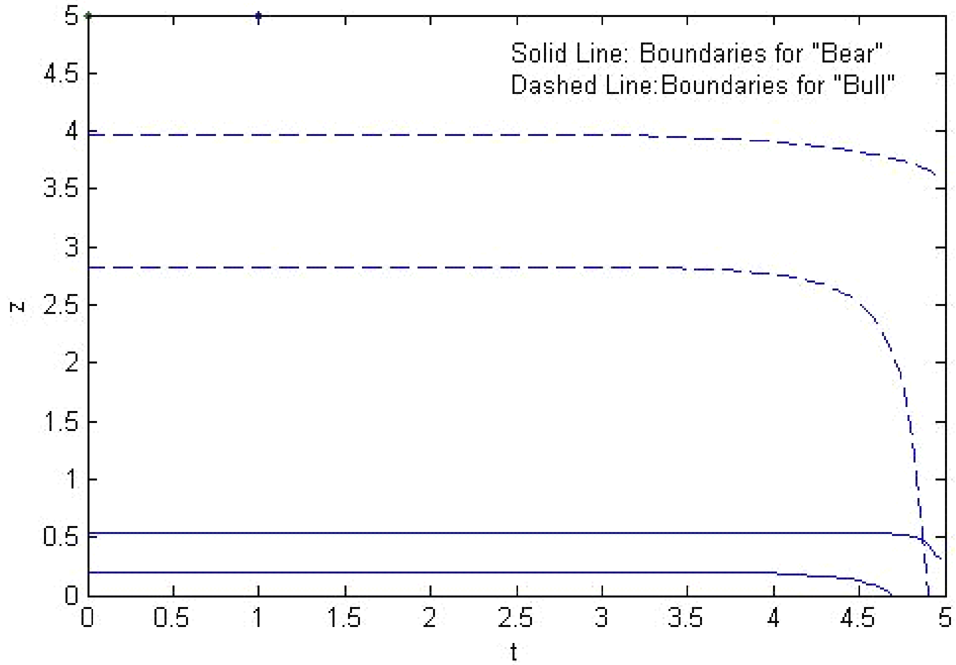

Figure 5.1: Plot of buying and selling boundaries for penalty method (4.2).

$dz = 0.01, dt = 0.025$.

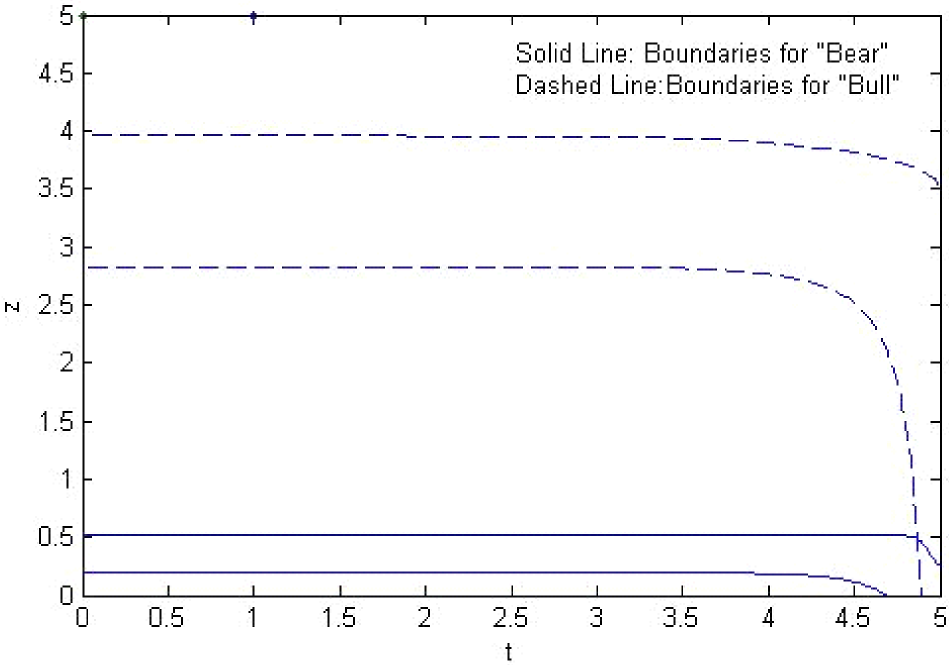

Figure 5.2: Plot of buying and selling boundaries for penalty method (4.2).

$dz = 0.016$, $dt = 0.01$.

5.2 The results of projected SOR method

We plot the optimal buying and selling boundaries in both bull market and bear market as functions of t using the above discretization in projected SOR method. We can see that the two methods are consistent using the same data as above. Notice that there are no penalty parameters here, and we use $\omega = 1.5$ in the iteration.

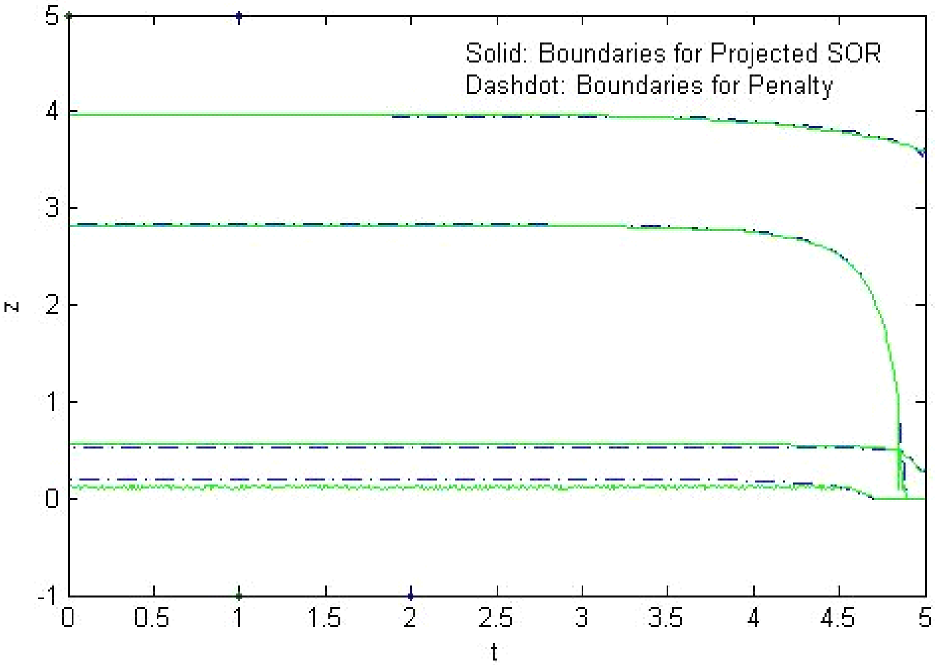

Figure 5.3: Plot of buying and selling boundaries for projected SOR method (4.3) and penalty method (4.2). $dz = 0.016$, $dt = 0.01$.

5.3 Changes in transaction costs

We plot the buying and selling boundaries for different λ and μ. We find that if transaction costs decrease, the buying boundaries shift upwards and the selling boundaries shift downwards, indicating that the investor is more encouraged to buy stocks when z is low and to sell stocks when z is high. This will lead to more frequent transactions. If transaction costs increase, the investor is discouraged to do tradings.

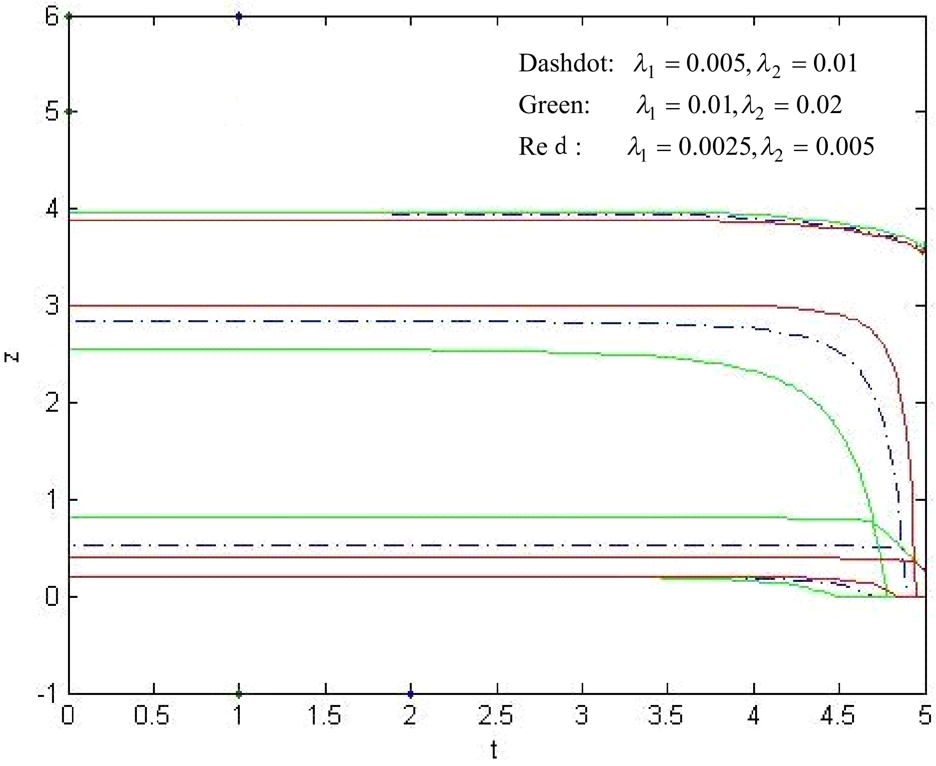

Figure 5.4: Plot of buying and selling boundaries for different transaction costs. Three cases: $\lambda = 0.01$, $\mu = 0.02$ and $\lambda = 0.005$, $\mu = 0.01$ and $\lambda = 0.0025$, $\mu = 0.005$.

5.4 Changes in rate of return of assets

If the rate of return of the risky asset increases, both the buying and selling boundaries will shift upwards. This is intuitive because the risky asset will become more attractive so that the wealth held in the risky asset will increase. The shifts in the boundaries become small as time to maturity declines, indicating that investors with longer expected time horizon are more sensitive to changes in rate of return of the risky asset. For example, younger investors are more sensitive to changes in rate of return. Notice that the NT regions become narrower as rate of return of risky asset increases, indicating that the investor is willing to transact more frequently disregarding the effects of transaction costs.

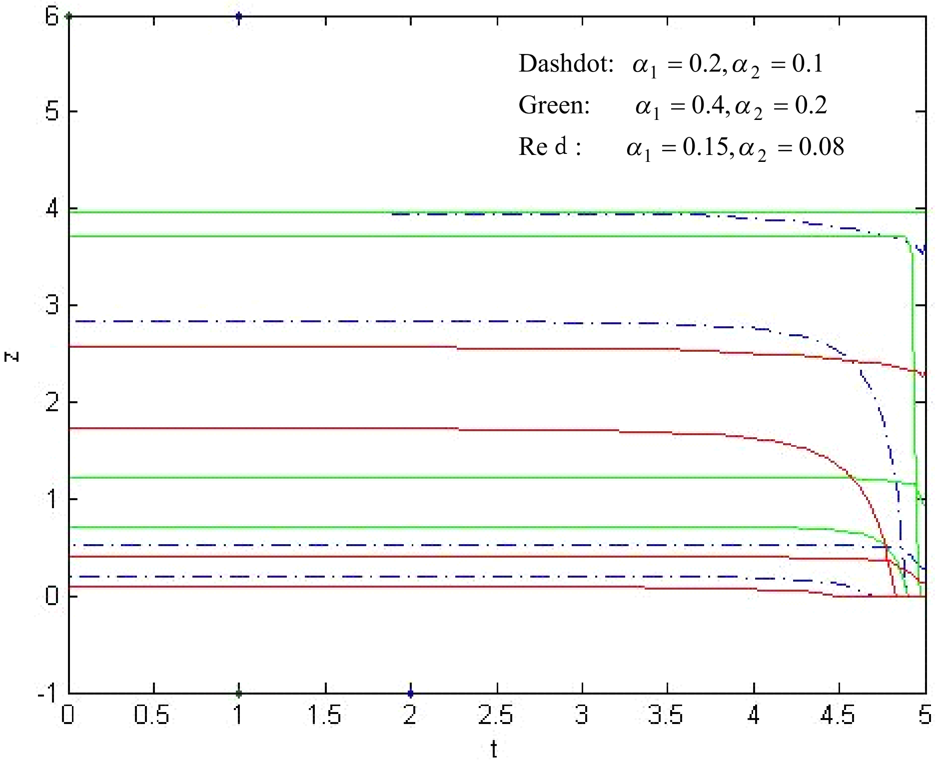

Figure 5.5: Plot of buying and selling boundaries for different rate of return of the risky asset. Three cases: $\alpha_1 = 0.4$, $\alpha_2 = 0.2$ and $\alpha_1 = 0.2, \alpha_2 = 0.1$ and $\alpha_1 = 0.15, \alpha_2 = 0.08$.

5.5 Changes in switch intensities

We first look at the case when the switch intensity from "bear" to "bull", k_2 , becomes high. We observe that the selling boundary in the bear market shifts upwards. This shows that the investor is not so hurried to sell the risky asset in bear market because switch probability from bear to bull becomes larger.

Next we look at the case when the switch intensity from "bull" to "bear", k_1 , becomes high. We observe that both the buying and selling boundaries in the bull market shift downwards. This shows that the investor is not so willing to hold much risky asset in bull market since switch probability from bull to bear is large. These observations are agree with the intuition.

At last, we look at the case when both switch intensities are low. We observe that the investor should keep a high fraction of wealth in stock in bull market and a low fraction in bear market.

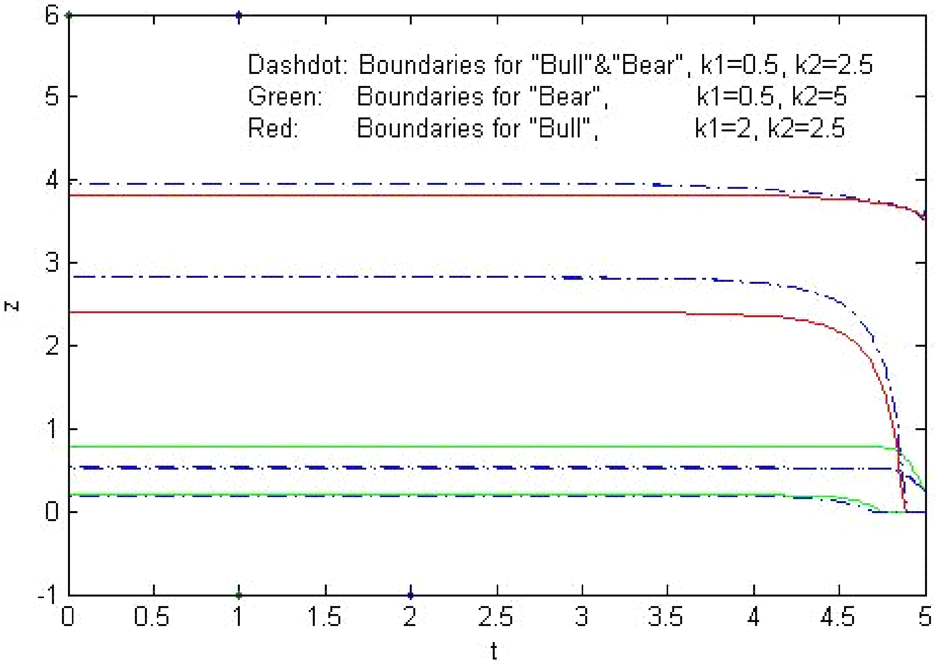

Figure 5.6: Plot of buying and selling boundaries for different switch intensities.

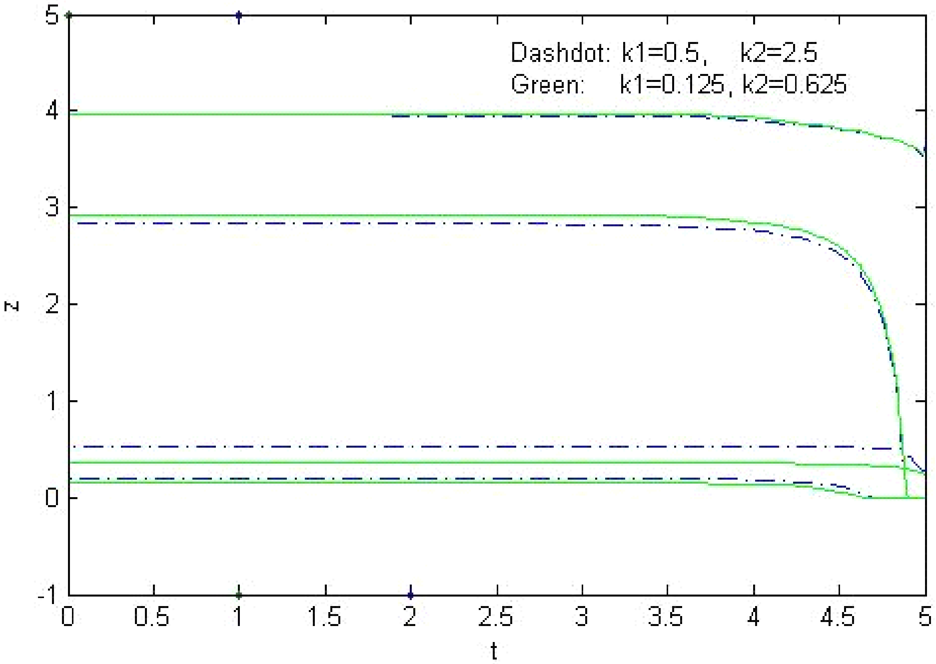

Figure 5.7: Plot of buying and selling boundaries for different switch intensities.

5.6 Changes in risk aversion index

We plot the buying and selling boundaries for different β. As β disappears after transformation, we need to plot for y instead of z. We find that if β increases, the buying and selling boundaries shift downwards, indicating that the investor is discouraged to hold stocks when he or she becomes more risk-averse.

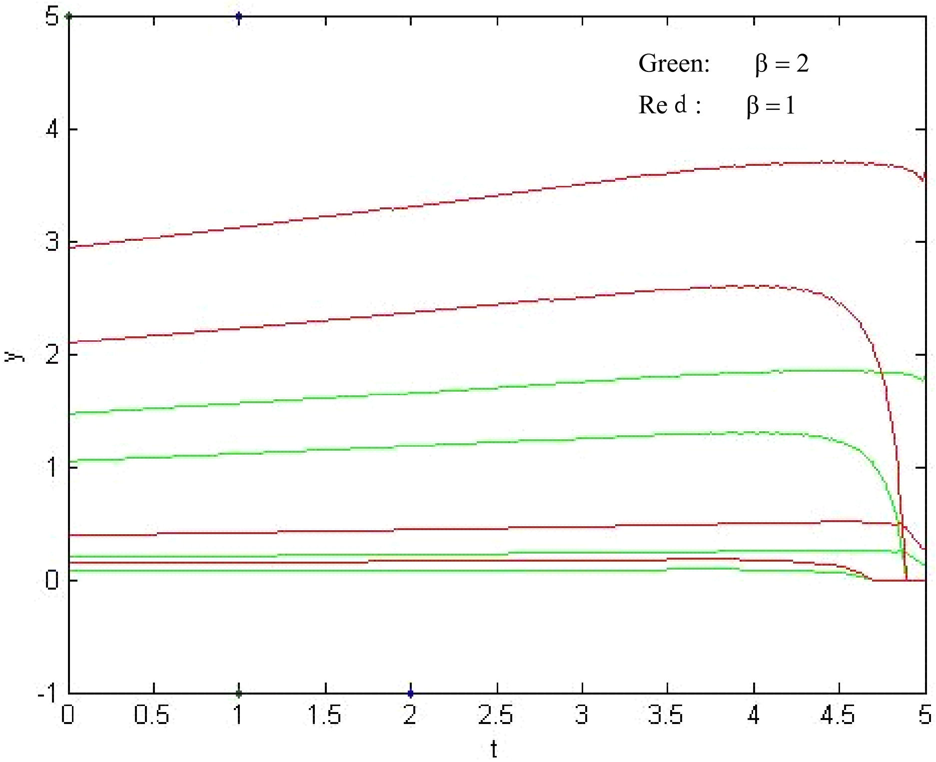

Figure 5.8: Plot of buying and selling boundaries for different risk aversion index.

5.7 Exponential Utility vs Logarithm Utility

In Dai et. al. (2010), the risk premium $\theta_i = \alpha_i - r - \sigma_i^2$ is defined in the logarithm utility case. However, in our exponential utility case, we consider the term $\alpha_i - r$ instead, which is always positive. Both the buying and selling boundaries in regime i will shift upwards if $\alpha_i - r$ increases. Moreover, the boundaries in exponential utility case are always positive. They are only related with the values in stocks. This is different from the logarithm utility case, in which the boundaries may be negative (leverage) and related with the values in stocks and bank account. There are also some threshold values of the parameters which help to examine the leverage problem in the logarithm utility case. This is not available for the exponential utility case now. But the intuitive results are similar in both cases, such as the decreasing of transaction costs will lead to more frequent transactions.

Chapter 6

Conclusion

We have presented numerical solutions to the finite horizon optimal investment problem with proportional transaction costs in a regime switching market. The exponential utility function has been considered. Previous studies have only provided numerical solutions to one-state market or logarithm and power utility cases in regime switching model. Following Dai et. al. (2010), we derived the HJB equations governing the value function and then solved the problem by penalty method and projected SOR method with fully implicit finite different scheme. In our exponential utility case, the problem is not equivalent to a parabolic double obstacle problem, and we need to solve the system of variational inequalities with gradient constraints directly.

Through the numerical results presented in chapter 5, we could see the above two methods are consistent in deriving the buying and selling boundaries. The main differences between these two methods are how to deal with the gradient constraints. We used penalty terms for gradient constraints to solve the whole system in penalty method while we put the constraints into the comparison conditions in the iteration of projected SOR method. After we saw the results of the two methods are reliable, we examined the properties of buying and selling boundaries of both bull and bear market when parameters are changed. The results are reasonable and intuitive. However, it still remains to study the leverage problem in exponential utility case with more mathematical analysis. We leave this as a topic for further research.

Appendix A

Source Code: Penalty Method

SwitchingExponential.m

```
% Fully implicit scheme to solve investment problem in two regimes
% Exponential Utility

% Parameters

T = 5;
Nt = 200;
Nu = 399;
dt = T/Nt;
du = 0.01;
alpha1 = 0.2;
sigma1 = 0.2;
r = 0.06;
alpha2 = 0.1;
sigma2 = 0.4;
k1 = 0.5;
k2 = 2.5;
beta = 100000;
lamda = 0.005;
mu = 0.01;

v1(1:Nu,:)=0;
v2(1:Nu,:)=0;
% Terminal Conditions
```

```
for i = 1 : Nu
	v1(i) = (1-mu)*i*du;
	v2(i) = (1-mu)*i*du;
end

A1 = zeros(Nu,Nu);
A2 = zeros(Nu,Nu);

for i = 2 : Nu-1
	A1(i,i-1)=
-0.5*dt*(sigma1^2)*(i^2)-0.5*dt*(sigma1^2)*(i^2)*(v1(i)-v1(i-1))-beta*dt/du*((1-m
u)>(v1(i)-v1(i-1))/du);
	A1(i,i)=
1+(alpha1-r)*i*dt+(sigma1^2)*(i^2)*dt+0.5*dt*(sigma1^2)*(i^2)*(v1(i)-v1(i-1))+bet
a*dt/du*((1-mu)>(v1(i)-v1(i-1))/du)+beta*dt/du*((v1(i+1)-v1(i))/du>(1+lamda));
	A1(i,i+1) =
(r-alpha1)*i*dt-0.5*dt*(sigma1^2)*(i^2)-beta*dt/du*((v1(i+1)-v1(i))/du>(1+lamda));
	A2(i,i-1) =
-0.5*dt*(sigma2^2)*(i^2)-0.5*dt*(sigma2^2)*(i^2)*(v2(i)-v2(i-1))-beta*dt/du*((1-m
u)>(v2(i)-v2(i-1))/du);
	A2(i,i) =
1+(alpha2-r)*i*dt+(sigma2^2)*(i^2)*dt+0.5*dt*(sigma2^2)*(i^2)*(v2(i)-v2(i-1))+bet
a*dt/du*((1-mu)>(v2(i)-v2(i-1))/du)+beta*dt/du*((v2(i+1)-v2(i))/du>(1+lamda));
	A2(i,i+1) =
(r-alpha2)*i*dt-0.5*dt*(sigma2^2)*(i^2)-beta*dt/du*((v2(i+1)-v2(i))/du>(1+lamda));
end
A1(1,1) =
1+(alpha1-r)*dt+0.5*(sigma1^2)*dt+beta*dt/du*((v1(2)-v1(1))/du>(1+lamda));
A1(1,2) =
(r-alpha1)*dt-0.5*dt*(sigma1^2)-beta*dt/du*((v1(2)-v1(1))/du>(1+lamda));
A1(Nu,Nu-1) =
-0.5*dt*(sigma1^2)*(Nu^2)-0.5*dt*(sigma1^2)*(Nu^2)*(v1(Nu)-v1(Nu-1))-beta*dt/
du*((1-mu)>(v1(Nu)-v1(Nu-1))/du);
A1(Nu,Nu) =
1+0.5*dt*(sigma1^2)*(Nu^2)+0.5*dt*(sigma1^2)*(Nu^2)*(v1(Nu)-v1(Nu-1))+beta*
dt/du*((1-mu)>(v1(Nu)-v1(Nu-1))/du);
A2(1,1) =
1+(alpha2-r)*dt+0.5*(sigma2^2)*dt+beta*dt/du*((v2(2)-v2(1))/du>(1+lamda));
A2(1,2) =
(r-alpha2)*dt-0.5*dt*(sigma2^2)-beta*dt/du*((v2(2)-v2(1))/du>(1+lamda));
A2(Nu,Nu-1) =
-0.5*dt*(sigma2^2)*(Nu^2)-0.5*dt*(sigma2^2)*(Nu^2)*(v2(Nu)-v2(Nu-1))-beta*dt/
du*((1-mu)>(v2(Nu)-v2(Nu-1))/du);
A2(Nu,Nu) =
```

```
1+0.5*dt*(sigma2^2)*(Nu^2)+0.5*dt*(sigma2^2)*(Nu^2)*(v2(Nu)-v2(Nu-1))+beta*
dt/du*((1-mu)>(v2(Nu)-v2(Nu-1))/du);

B1 = zeros(Nu,1);
B2 = zeros(Nu,1);
for i = 2 : Nu-1
    B1(i) =
-k1*dt*(1-exp(v1(i)-v2(i)))-beta*dt*(1-mu)*((1-mu)>(v1(i)-v1(i-1))/du)+beta*dt*(1+
lamda)*((v1(i+1)-v1(i))/du>(1+lamda));
    B2(i) =
-k2*dt*(1-exp(v2(i)-v1(i)))-beta*dt*(1-mu)*((1-mu)>(v2(i)-v2(i-1))/du)+beta*dt*(1+
lamda)*((v2(i+1)-v2(i))/du>(1+lamda));
end
B1(1) =
-k1*dt*(1-exp(v1(1)-v2(1)))+beta*dt*(1+lamda)*((v1(2)-v1(1))/du>(1+lamda))+0.5*
(sigma1^2)*dt*(1+lamda)*du+0.5*(sigma1^2)*dt*(1+lamda)^2*(du^2);
B1(Nu)=
-k1*dt*(1-exp(v1(Nu)-v2(Nu)))-beta*dt*(1-mu)*((1-mu)>(v1(Nu)-v1(Nu-1))/du)-Nu
*(alpha1-r)*(1-mu)*du*dt-0.5*dt*(sigma1^2)*(Nu^2)*du*(1-mu);
B2(1) =
-k2*dt*(1-exp(v2(1)-v1(1)))+beta*dt*(1+lamda)*((v2(2)-v2(1))/du>(1+lamda))+0.5*
(sigma2^2)*dt*(1+lamda)*du+0.5*(sigma2^2)*dt*(1+lamda)^2*(du^2);
B2(Nu)=
-k2*dt*(1-exp(v2(Nu)-v1(Nu)))-beta*dt*(1-mu)*((1-mu)>(v2(Nu)-v2(Nu-1))/du)-Nu
*(alpha2-r)*(1-mu)*du*dt-0.5*dt*(sigma2^2)*(Nu^2)*du*(1-mu);

BR1=zeros(Nt,1);
BR2=zeros(Nt,1);
SR1=zeros(Nt,1);
SR2=zeros(Nt,1);

for j = 1 : Nt
    vector1 = v1;
    vector2 = v2;

    vv1 = A1\(v1-B1);
    vv2 = A2\(v2-B2);

    count = 0;

    while norm(vector1-vv1)/norm(vector1) > 0.00001 ||
         norm(vector2-vv2)/norm(vector2) > 0.00001
        vector1 = vv1;
        vector2 = vv2;
```

```
        for i = 2 : Nu-1
            A1(i,i-1) =
-0.5*dt*(sigma1^2)*(i^2)-0.5*dt*(sigma1^2)*(i^2)*(vector1(i)-vector1(i-1))-beta*dt/
du*((1-mu)>(vector1(i)-vector1(i-1))/du);
            A1(i,i) =
1+(alpha1-r)*i*dt+(sigma1^2)*(i^2)*dt+0.5*dt*(sigma1^2)*(i^2)*(vector1(i)-vector1
(i-1))+beta*dt/du*((1-mu)>(vector1(i)-vector1(i-1))/du)+beta*dt/du*((vector1(i+1)-v
ector1(i))/du>(1+lamda));
            A1(i,i+1) =
(r-alpha1)*i*dt-0.5*dt*(sigma1^2)*(i^2)-beta*dt/du*((vector1(i+1)-vector1(i))/du>(1
+lamda));
            A2(i,i-1) =
-0.5*dt*(sigma2^2)*(i^2)-0.5*dt*(sigma2^2)*(i^2)*(vector2(i)-vector2(i-1))-beta*dt/
du*((1-mu)>(vector2(i)-vector2(i-1))/du);
            A2(i,i) =
1+(alpha2-r)*i*dt+(sigma2^2)*(i^2)*dt+0.5*dt*(sigma2^2)*(i^2)*(vector2(i)-vector2
(i-1))+beta*dt/du*((1-mu)>(vector2(i)-vector2(i-1))/du)+beta*dt/du*((vector2(i+1)-v
ector2(i))/du>(1+lamda));
            A2(i,i+1) =
(r-alpha2)*i*dt-0.5*dt*(sigma2^2)*(i^2)-beta*dt/du*((vector2(i+1)-vector2(i))/du>(1
+lamda));
        end
        A1(1,1) =
1+(alpha1-r)*dt+0.5*(sigma1^2)*dt+beta*dt/du*((vector1(2)-vector1(1))/du>(1+lam
da));
        A1(1,2) =
(r-alpha1)*dt-0.5*dt*(sigma1^2)-beta*dt/du*((vector1(2)-vector1(1))/du>(1+lamda));
        A1(Nu,Nu-1) =
-0.5*dt*(sigma1^2)*(Nu^2)-0.5*dt*(sigma1^2)*(Nu^2)*(vector1(Nu)-vector1(Nu-1))
-beta*dt/du*((1-mu)>(vector1(Nu)-vector1(Nu-1))/du);
        A1(Nu,Nu) =
1+0.5*dt*(sigma1^2)*(Nu^2)+0.5*dt*(sigma1^2)*(Nu^2)*(vector1(Nu)-vector1(Nu-
1))+beta*dt/du*((1-mu)>(vector1(Nu)-vector1(Nu-1))/du);
        A2(1,1) =
1+(alpha2-r)*dt+0.5*(sigma2^2)*dt+beta*dt/du*((vector2(2)-vector2(1))/du>(1+lam
da));
        A2(1,2) =
(r-alpha2)*dt-0.5*dt*(sigma2^2)-beta*dt/du*((vector2(2)-vector2(1))/du>(1+lamda));
        A2(Nu,Nu-1) =
-0.5*dt*(sigma2^2)*(Nu^2)-0.5*dt*(sigma2^2)*(Nu^2)*(vector2(Nu)-vector2(Nu-1))
-beta*dt/du*((1-mu)>(vector2(Nu)-vector2(Nu-1))/du);
        A2(Nu,Nu) =
1+0.5*dt*(sigma2^2)*(Nu^2)+0.5*dt*(sigma2^2)*(Nu^2)*(vector2(Nu)-vector2(Nu-
1))+beta*dt/du*((1-mu)>(vector2(Nu)-vector2(Nu-1))/du);
```

```
            B1 = zeros(Nu,1);
            B2 = zeros(Nu,1);
            for i = 2 : Nu-1
                  B1(i) =
-k1*dt*(1-exp(vector1(i)-vector2(i)))-beta*dt*(1-mu)*((1-mu)>(vector1(i)-vector1(i-1))/du)+beta*dt*(1+lamda)*((vector1(i+1)-vector1(i))/du>(1+lamda));
                  B2(i) =
-k2*dt*(1-exp(vector2(i)-vector1(i)))-beta*dt*(1-mu)*((1-mu)>(vector2(i)-vector2(i-1))/du)+beta*dt*(1+lamda)*((vector2(i+1)-vector2(i))/du>(1+lamda));
            end
            B1(1) =
-k1*dt*(1-exp(vector1(1)-vector2(1)))+beta*dt*(1+lamda)*((vector1(2)-vector1(1))/du>(1+lamda))+0.5*(sigma1^2)*dt*(1+lamda)*du+0.5*(sigma1^2)*dt*(1+lamda)^2*(du^2);
            B1(Nu)=
-k1*dt*(1-exp(vector1(Nu)-vector2(Nu)))-beta*dt*(1-mu)*((1-mu)>(vector1(Nu)-vector1(Nu-1))/du)-Nu*(alpha1-r)*(1-mu)*du*dt-0.5*(Nu^2)*(sigma1^2)*(1-mu)*du*dt;
            B2(1) =
-k2*dt*(1-exp(vector2(1)-vector1(1)))+beta*dt*(1+lamda)*((vector2(2)-vector2(1))/du>(1+lamda))+0.5*(sigma2^2)*dt*(1+lamda)*du+0.5*(sigma2^2)*dt*(1+lamda)^2*(du^2);
            B2(Nu)=
-k2*dt*(1-exp(vector2(Nu)-vector1(Nu)))-beta*dt*(1-mu)*((1-mu)>(vector2(Nu)-vector2(Nu-1))/du)-Nu*(alpha2-r)*(1-mu)*du*dt-0.5*(Nu^2)*(sigma2^2)*(1-mu)*du*dt;

            vv1 = A1\(v1-B1);
            vv2 = A2\(v2-B2);
            count = count+1;
            %if count == 100
             %     break;
            %end
      end
      v1 = vv1;
      v2 = vv2;

      flag1=0;
      flag2=0;
      flag3=0;
      flag4=0;

      for k = 2 : Nu-1
```

```
        if   (v1(k)-v1(k-1))/du >= (1+lamda)
              BR1(j) = (k-1)*du;
        end
        if   (v1(k+1)-v1(k))/du > (1-mu)
              SR1(j) = k*du;
        end
        if   (v2(k)-v2(k-1))/du >= (1+lamda)
              BR2(j) = (k-1)*du;
        end
        if   (v2(k+1)-v2(k))/du > (1-mu)
              SR2(j) = k*du;
        end
        if flag1 & flag2 & flag3 & flag4
              break;
        end
    end
end
plot((T-dt:-dt:0),SR1,'--')
hold on
plot((T-dt:-dt:0),BR1,'-')
hold on
plot((T-dt:-dt:0),SR2,':')
hold on
plot((T-dt:-dt:0),BR2,'-.')
```

Appendix B

Source Code: Projected SOR Method

ExponentialSOR.m

```
% Fully implicit scheme to solve investment problem in two regimes
% Exponential Utility

% Parameters

T = 5;
Nt = 200;
Nu = 399;
dt = T/Nt;
du = 0.01;
alpha1 = 0.2;
sigma1 = 0.2;
r = 0.06;
alpha2 = 0.1;
sigma2 = 0.4;
k1 = 0.5;
k2 = 2.5;
lamda = 0.005;
mu = 0.01;
omega = 1.5;

v1(1:Nu,:)=0;
v2(1:Nu,:)=0;
% Terminal Conditions
```

```
for i = 1 : Nu
    v1(i) = (1-mu)*i*du;
    v2(i) = (1-mu)*i*du;
end

A1 = zeros(Nu,Nu);
A2 = zeros(Nu,Nu);

for i = 2 : Nu-1
    A1(i,i-1) = -0.5*dt*(sigma1^2)*(i^2)-0.5*dt*(sigma1^2)*(i^2)*(v1(i)-v1(i-1));
    A1(i,i)     =
1+(alpha1-r)*i*dt+(sigma1^2)*(i^2)*dt+0.5*dt*(sigma1^2)*(i^2)*(v1(i)-v1(i-1));
    A1(i,i+1) = (r-alpha1)*i*dt-0.5*dt*(sigma1^2)*(i^2);
    A2(i,i-1) = -0.5*dt*(sigma2^2)*(i^2)-0.5*dt*(sigma2^2)*(i^2)*(v2(i)-v2(i-1));
    A2(i,i)     =
1+(alpha2-r)*i*dt+(sigma2^2)*(i^2)*dt+0.5*dt*(sigma2^2)*(i^2)*(v2(i)-v2(i-1));
    A2(i,i+1) = (r-alpha2)*i*dt-0.5*dt*(sigma2^2)*(i^2);
end
A1(1,1) = 1+(alpha1-r)*dt+0.5*(sigma1^2)*dt;
A1(1,2) = (r-alpha1)*dt-0.5*dt*(sigma1^2);
A1(Nu,Nu-1) =
-0.5*dt*(sigma1^2)*(Nu^2)-0.5*dt*(sigma1^2)*(Nu^2)*(v1(Nu)-v1(Nu-1));
A1(Nu,Nu) =
1+0.5*dt*(sigma1^2)*(Nu^2)+0.5*dt*(sigma1^2)*(Nu^2)*(v1(Nu)-v1(Nu-1));
A2(1,1) = 1+(alpha2-r)*dt+0.5*(sigma2^2)*dt;
A2(1,2) = (r-alpha2)*dt-0.5*dt*(sigma2^2);
A2(Nu,Nu-1) =
-0.5*dt*(sigma2^2)*(Nu^2)-0.5*dt*(sigma2^2)*(Nu^2)*(v2(Nu)-v2(Nu-1));
A2(Nu,Nu) =
1+0.5*dt*(sigma2^2)*(Nu^2)+0.5*dt*(sigma2^2)*(Nu^2)*(v2(Nu)-v2(Nu-1));

B1 = zeros(Nu,1);
B2 = zeros(Nu,1);
for i = 2 : Nu-1
    B1(i) = -k1*dt*(1-exp(v1(i)-v2(i)));
    B2(i) = -k2*dt*(1-exp(v2(i)-v1(i)));
end
B1(1) =
-k1*dt*(1-exp(v1(1)-v2(1)))+0.5*(sigma1^2)*dt*(1+lamda)*du+0.5*(sigma1^2)*dt*
(1+lamda)^2*(du^2);
B1(Nu)=
-k1*dt*(1-exp(v1(Nu)-v2(Nu)))-Nu*(alpha1-r)*(1-mu)*du*dt-0.5*dt*(sigma1^2)*(N
u^2)*du*(1-mu);
B2(1) =
```

```
-k2*dt*(1-exp(v2(1)-v1(1)))+0.5*(sigma2^2)*dt*(1+lamda)*du+0.5*(sigma2^2)*dt*
(1+lamda)^2*(du^2);
B2(Nu)=
-k2*dt*(1-exp(v2(Nu)-v1(Nu)))-Nu*(alpha2-r)*(1-mu)*du*dt-0.5*dt*(sigma2^2)*(N
u^2)*du*(1-mu);

BR1=zeros(Nt,1);
BR2=zeros(Nt,1);
SR1=zeros(Nt,1);
SR2=zeros(Nt,1);

for j = 1 : Nt
    vector1 = v1;
    vector2 = v2;

    vv1 = A1\(v1-B1);
    vv2 = A2\(v2-B2);

    count = 0;

    while norm(vector1-vv1)/norm(vector1) > 0.00001 ||
norm(vector2-vv2)/norm(vector2) > 0.00001
        vector1 = vv1;
        vector2 = vv2;
        for i = 2 : Nu-1
            A1(i,i-1) =
-0.5*dt*(sigma1^2)*(i^2)-0.5*dt*(sigma1^2)*(i^2)*(vector1(i)-vector1(i-1));
            A1(i,i)     =
1+(alpha1-r)*i*dt+(sigma1^2)*(i^2)*dt+0.5*dt*(sigma1^2)*(i^2)*(vector1(i)-vector1
(i-1));
            A1(i,i+1) = (r-alpha1)*i*dt-0.5*dt*(sigma1^2)*(i^2);
            A2(i,i-1) =
-0.5*dt*(sigma2^2)*(i^2)-0.5*dt*(sigma2^2)*(i^2)*(vector2(i)-vector2(i-1));
            A2(i,i)     =
1+(alpha2-r)*i*dt+(sigma2^2)*(i^2)*dt+0.5*dt*(sigma2^2)*(i^2)*(vector2(i)-vector2
(i-1));
            A2(i,i+1) = (r-alpha2)*i*dt-0.5*dt*(sigma2^2)*(i^2);
        end
        A1(1,1) = 1+(alpha1-r)*dt+0.5*(sigma1^2)*dt;
        A1(1,2) = (r-alpha1)*dt-0.5*dt*(sigma1^2);
        A1(Nu,Nu-1) =
-0.5*dt*(sigma1^2)*(Nu^2)-0.5*dt*(sigma1^2)*(Nu^2)*(vector1(Nu)-vector1(Nu-1))
;
        A1(Nu,Nu) =
```

```
1+0.5*dt*(sigma1^2)*(Nu^2)+0.5*dt*(sigma1^2)*(Nu^2)*(vector1(Nu)-vector1(Nu-1));
            A2(1,1) = 1+(alpha2-r)*dt+0.5*(sigma2^2)*dt;
            A2(1,2) = (r-alpha2)*dt-0.5*dt*(sigma2^2);
            A2(Nu,Nu-1) =
-0.5*dt*(sigma2^2)*(Nu^2)-0.5*dt*(sigma2^2)*(Nu^2)*(vector2(Nu)-vector2(Nu-1));
            A2(Nu,Nu) =
1+0.5*dt*(sigma2^2)*(Nu^2)+0.5*dt*(sigma2^2)*(Nu^2)*(vector2(Nu)-vector2(Nu-1));

            B1 = zeros(Nu,1);
            B2 = zeros(Nu,1);
            for i = 2 : Nu-1
                  B1(i) = -k1*dt*(1-exp(vector1(i)-vector2(i)));
                  B2(i) = -k2*dt*(1-exp(vector2(i)-vector1(i)));
            end
            B1(1) =
-k1*dt*(1-exp(vector1(1)-vector2(1)))+0.5*(sigma1^2)*dt*(1+lamda)*du+0.5*(sigma1^2)*dt*(1+lamda)^2*(du^2);
            B1(Nu)=
-k1*dt*(1-exp(vector1(Nu)-vector2(Nu)))-Nu*(alpha1-r)*(1-mu)*du*dt-0.5*(Nu^2)*(sigma1^2)*(1-mu)*du*dt;
            B2(1) =
-k2*dt*(1-exp(vector2(1)-vector1(1)))+0.5*(sigma2^2)*dt*(1+lamda)*du+0.5*(sigma2^2)*dt*(1+lamda)^2*(du^2);
            B2(Nu)=
-k2*dt*(1-exp(vector2(Nu)-vector1(Nu)))-Nu*(alpha2-r)*(1-mu)*du*dt-0.5*(Nu^2)*(sigma2^2)*(1-mu)*du*dt;

            vv1(1) =
omega*((-A1(1,2)*vector1(2)+v1(1)-B1(1))/A1(1,1))-(omega-1)*vector1(1);
            vv2(1) =
omega*((-A2(1,2)*vector2(2)+v2(1)-B2(1))/A2(1,1))-(omega-1)*vector2(1);
            for i = 2 : Nu-1
                  vv1(i) =
omega*((-A1(i,i-1)*vv1(i-1)-A1(i,i+1)*vector1(i+1)+v1(i)-B1(i))/A1(i,i))-(omega-1)*vector1(i);
                  vv2(i) =
omega*((-A2(i,i-1)*vv2(i-1)-A2(i,i+1)*vector2(i+1)+v2(i)-B2(i))/A2(i,i))-(omega-1)*vector2(i);
            end
            vv1(Nu) =
omega*((-A1(Nu,Nu-1)*vv1(Nu-1)+v1(Nu)-B1(Nu))/A1(Nu,Nu))-(omega-1)*vector
```

```
1(Nu);
        vv2(Nu) = 
omega*((-A2(Nu,Nu-1)*vv2(Nu-1)+v2(Nu)-B2(Nu))/A2(Nu,Nu))-(omega-1)*vector
2(Nu);
    end
        C1 = vv1;
        C2 = vv2;
        for i = 2 : Nu
            if C1(i) > C1(i-1)+(1+lamda)*du
                vv1(i) = vv1(i-1)+(1+lamda)*du;
            end
            if C1(i) < C1(i-1)+(1-mu)*du
                vv1(i) = vv1(i-1)+(1-mu)*du;
            end
            if C1(i) <= C1(i-1)+(1+lamda)*du && C1(i) >= C1(i-1)+(1-mu)*du
                vv1(i) = vv1(i-1)+C1(i)-C1(i-1);
            end
            if C2(i) > C2(i-1)+(1+lamda)*du
                vv2(i) = vv2(i-1)+(1+lamda)*du;
            end
            if C2(i) < C2(i-1)+(1-mu)*du
                vv2(i) = vv2(i-1)+(1-mu)*du;
            end
            if C2(i) <= C2(i-1)+(1+lamda)*du && C2(i) >= C2(i-1)+(1-mu)*du
                vv2(i) = vv2(i-1)+C2(i)-C2(i-1);
            end
        end


        count = count+1;
        %if count == 100
         %     break;
        %end
    v1 = vv1;
    v2 = vv2;

    flag1=0;
    flag2=0;
    flag3=0;
    flag4=0;

    for k = 2 : Nu-1
        if    (v1(k)-v1(k-1)) >= (1+lamda)*du
            BR1(j) = (k-1)*du;
```

```
        end
        if   (v1(k+1)-v1(k)) > (1-mu)*du+0.0000001
             SR1(j) = k*du+0.1;
        end
        if   (v2(k)-v2(k-1)) >= (1+lamda)*du
             BR2(j) = (k-1)*du;
        end
        if   (v2(k+1)-v2(k)) > (1-mu)*du+0.0000001
             SR2(j) = k*du;
        end
        if flag1 & flag2 & flag3 & flag4
             break;
        end
    end
end
SR2(1)=0.3;
plot((T-dt:-dt:0),SR1,'--')
hold on
plot((T-dt:-dt:0),BR1,'-')
hold on
plot((T-dt:-dt:0),SR2,':')
hold on
plot((T-dt:-dt:0),BR2,'-.')
```

Bibliography

[1] Chong, Kaiyun, 2006, Finite Horizon Optimal Investment and Consumption with Transaction Costs, Honours Thesis, Department of Mathematics, National University of Singapore.

[2] Dai, Min, and Fahuai Yi, 2009, Finite-Horizon Optimal Investment with Transaction Costs: A Parabolic Double Obstacle Problem, *Journal of Differential Equations*, 246, 1445-146.

[3] Dai, Min, Hefei Wang and Zhou Yang, 2010, Leverage Management in a Bull-Bear Switching Market, working paper.

[4] Dai, Min, and Yifei Zhong, 2010, Penalty Methods for Continuous-Time Portfolio Selection with Proportional Transaction Costs, *Journal of Computational Finance*, to appear.

[5] Davis, M. H. A., and A. R. Norman, 1990, Portfolio Selection with Transaction Costs, *Mathematics of Operations Research*, 15, 676-713.

[6] Forsyth, P. A., and K. R. Vetzal, 2004, Quadratic Convergence for Valuing American Options Using a Penalty Method, *SIAM J. Sci. Comp.*, 97(2). 321-352.

[7] Gennotte, G., and A. Jung, 1994, Investment Strategies under

Transaction Costs: The Finite Horizon Case, *Management Science*, 40, 385-404.

[8] Jang, Bong-Gyu, Hyeng Keun Koo, Hong Liu, and Mark Loewenstein, 2007, Liquidity Premia and Transaction Costs, *Journal of Finance* 62, 2329-2366.

[9] Li, Wen Yi, 2010, Portfolio Selection with Transaction Costs in a Regime Switching Model, Honours Thesis, Department of Mathematics, National University of Singapore.

[10] Liu, H., and M. Loewenstein, 2002, Optimal Portfolio Selection with Transaction Costs and Finite Horizons, *Review of Financial Studies*, 15(3), 805-835.

[11] Magill, M. J. P., and G. M. Constantinides, 1976, Portfolio Selection with Transaction Costs, *Journal of Economic Theory*, 13, 264-271.

[12] Merton, Robert C., 1969, Lifetime Portfolio Selection under Uncertainty: The Continuous-Time Case, *Review of Economics and Statistics*, 51(3), 247-257.

[13] Merton, Robert C., 1971, Optimum Consumption and Portfolio Rules in a Continuous-Time Model, *Journal of Economic Theory*, 3, 373-413.

[14] Uichanco, J., 2006, Numerical Algorithms for Optimal Consumption and Investment with Transaction Costs, Honours Thesis, Department of Mathematics, National University of Singapore.